INTRODUCTION

Welcome back. I want to thank Dr. McCarty for sharing with us his knowledge and insights on the trends and progress being made to rollout open source software.

OUTLINE OF PRESENTATION

SLIDE 1

This segment is intended to simply relate the experiences of the Air Resources Board in using the Open Source tools and software Dr. McCarty talked about. I have to admit at the outset: we are fans.

The main points I want to cover are:

1) ARB's IT Program and Environment

2) ARB's Use of Open Source Software

3) A Case Study

4) ARB's Open Source Directions

5) Open Source Futures

ARB'S IT PROGRAM AND ENVIRONMENT

SLIDE 2

Before we get too far along, I'd like to characterize ARB's IT program and management culture.

First, like yours, ARB's IT program is conservative. The systems we develop must be useful, reliable, timely, accurate, yet reasonably easy to integrate with the "outside" world. Given the nature of today's budget, the systems must also be durable and efficient.

Secondly, ARB's IT program has a high level of executive visibility. As CIO, I report directly to the executive office. Our Office of Information Security is centralized, ….as is our project management office. The executive office strongly supports effective project management for IT as well as for non-IT efforts.

Briefly, ARB's IT organization is comprised of 50 professional staff, supporting 1,200 employees, divided into seven divisions, located in two locations: Sacramento and El Monte. It's responsible for systems development, ARB's Internet and Intranet services, office systems,

ARB's high speed network, enterprise servers as well as ARB's telephone infrastructure.

With regards to ARB's management style: Simply put, ARB puts a lot of trust and confidence in its staff. It encourages creativity and innovation at all levels. In my experience here, new ideas continually arise from all corners of the organization.

ARB's management environment is well suited for adapting open source solutions. Management tends to believe that not all great or elegant solutions, IT or otherwise, need to be expensive, must come pre-packaged or shrink-wrapped, or include every bell and whistle. The goal is to facilitate and enhance individual productivity, albeit at a reasonable cost.

ARB JOINS INTERNET

The ARB's IT program in many ways has paralleled that of the Internet itself. We joined the Internet movement in 1991. Working with Teale Data Center… we built the Ethernet infrastructure needed to connect our air quality modellers with the San Diego SuperComputer. We also introduced Internet-based email to the Board, using products like Eudoria, Pegasus.

Those were primal days of the Internet. Some of you will recall using Internet "search engines" with names like Gopher, Archie, Veronica, names taken from Archie Comic Books.

ARB'S JOINS WORLD WIDE WEB

Our World Wide Web services program began in 1994, when the Web only sported 50 servers. Today there are over 35 million.

At that time, we experimented with delivering ARB information to the public and exchanging data with our districts via an electronic bulletin board.   We also tested open source WEB products, like Slackware's distribution of Linux, which included the original CERN Web server. This early version of Linux, however, was a touch too anemic for our needs.

We also tested Microsoft's NT 3.5.1, using a Web server called WEB-Site One, running on a PC386.   This combination worked; and became our first WEB service implementation. The cost was about $250.

A year later, however, Linux was much improved. We purchased a $50 Linux manual from Red Hat that included a CD in the flap. And installed this version of Linux to support a new proxy service.   The

Linux based proxy service protected us from hackers who were then

beginning to prey on NT installations.

We subsequently implemented Linux to support our list serv program,

ftp server, the network DNS, and our Internet search engine.

Beginning in 2000, as I will describe later in more detail, we began to develop Web-based applications using Linux as the OS, Apache as the WEB server, and PHP as the scripting language. We migrated our NT Web servers to Linux at this time as well.

The net result of these efforts was measureable in cost savings. We also greatly increased our understanding of Internet systems, and benefited from inexpensive redundancy, systems reliability, freedom from vendor licensing strategies, and increased control over operations.   Our experiences confirmed what the trade magazine had been saying about these open source products.

SLIDE 6

A CASE STUDY

I'd like to share our experience in migrating an application from a commercial platform to one based largely on open source products.

In 1994, our emissions inventory systems group was developing an Oracle database on a local HP/UX server, and on a Sun Solaris server hosted at the Teale Data Center. The application was being built using Oracle's PL/SQL along with the Oracle Developer suite of tools. The design called for users to telnet to the appropriate server to access character mode menus, data entry forms, and reports.

By 1995, it appeared that the Web and Web browsers offered a better way to deliver applications to clients as well as provide information to the general public. So, we bought Oracle's Web Application Server, and used it to build ARB's first Web-based application, replacing the telnet interface of the emission inventory system.

SLIDE 7

Unfortunately, we found that Oracle's early tools for developing Web applications had, as they say…issues. Each new release added significantly different features. This led to stability problems: Staff had to relearn Oracle's ever-changing Web development tools and rewrite the applications.

Costs were also becoming a factor: each new "bigger and improved" version of Oracle's Web server and tools became more expensive. As server licensing costs increased, we found it more difficult to budget for second or third Web application servers; servers needed for testing, development, or fail over purposes.

Using the Oracle Web Application Server also meant that the application server worked only with Oracle databases. ARB had other databases in use, like DB2, SQL Server, and Microsoft Access. Oracle's Web application server was inoperable with these other systems.

SLIDE 8

By 1999 we started looking for replacement software which would be:
1) cheaper to acquire and maintain, 2) provide greater stability, 3)
provide better programming tools, and 4) provide greater inter-
operability.

Since we had experience using Linux/Apache to support our proxy
servers, ftp servers, DNS, and list servers, we looked for open source
products to replace the Oracle Web Application Server.

SLIDE 9

We found that Linux and Apache met our performance requirements, again.   As an aside, Apache software is now used on over 63% of all Web servers.

We still lacked an application scripting language, however, one to handle transactions. Actually, there were many open source scripting languages we could have used, like Perl, C, Python.   We happened upon PHP. PHP is a language developed specifically for developing Web applications. It also has connectors to a wide variety of databases, including Oracle, DB2, and others.

PHP also integrated well with Linux and Apache, and contained many functions which simplified building Web applications. We've since found that the PHP user/developer community refines and incrementally adds functionality to the language on a regular basis.

Our initial trials using Linux/Apache/PHP instead of Oracle's Web Application Server progressed smoothly. There were some

telecommunication issues, but we found user groups on the Internet

to be a rich source of information and assistance in resolving them.

In the end, while we kept the data base itself in Oracle, we replaced

the other components of the emissions inventory application with the

Linux, Apache, and PHP suite

SLIDE 10

Our programmers were able to integrate the functions easily, and they were impressed with the breadth of functionality this combination provided for developing Web applications.

The change to open source products was transparent to the users.

And security was not an issue since the data base and web application servers remained behind our Fire Wall; all public inquiries to the system are handled by the Proxy server.

SLIDE 11

This table compares what it would cost today to develop this emissions inventory system using Oracles's software versus open source software. The system runs on two uni-processor Intel-based servers.

	Proprietary Option	**Open Source**
Operating System, for 2 servers:	Windows 2000: $1,600	Linux: $59 (free if downloaded)
Web Appl Server, for 2 servers:	Oracle: $20,000	Apache: $0
App Dev Software, for 6 developers:	JDeveloper: $6,000	PHP: $0
Annual Maint Cost	Oracle: $2,500	$0
Training	Comparable	Comparable
Hardware	Comparable	Comparable

We estimate that our costs would be about $27,600: $1,600 for the OS; $20,000 for the Web App Server and $6,000 for the Development Software, for 6 programmers. In contrast, by using the open source products we only paid $59.   And we have no ongoing required Annual Maintenance Costs.

Further, with the open source solution, we can replicate as many

implementations of the system as necessary, not just for production,

and testing, but also for admin. training, spares for operational

recovery, and so on, at no cost, and with no worries over licensing

violations.

Future expansion costs: As we add Secure Socket Layer capability,

expand the use of Web applications, and add processors and

servers, proprietary costs will likely increase. On the other hand, we

anticipate that open source solution costs will remain low.

SLIDE 12

LIGHT THE LAMP

Since the migration to the Linux/Apache/PHP suite, we discovered

MySQL (*my sequel*), an open source product for developing data

bases. This product integrates well with the Linux/Apache/PHP suite.

We've built several small to medium sized systems with MySQL and

find that it easily competes with proprietary products on the market,

like MS ACCESS, MS SQL; even Oracle in some cases. It handles

JOINS quickly, as well as transactions and queries.

However, it's currently limited to a single server, although multiple

processors are supported. MySQL cannot do simultaneous

processing; and is not well suited for clustering.   But we expect these

limitations to change with time.

Not withstanding these issues, we have standardized on MySQL for

all small and most medium sized data base applications, where a

product like MS ACCESS, Oracle, would otherwise be used. Further,

we have converted nearly all of our small systems to MySQL.

In summary, ARB has now standardized on what the open source

community refers to as the LAMP suite: Linux, Apache, MySQL, PHP,

to develop and maintain a wide range of Web-based systems.

ARB OPEN SOURCE FUTURES

SLIDE 13, then 14

This slide summarizes our current commitment to Open Source

products:

Functions	Open Source Services
WEB Server Services	WEB Servers OS*: Linux* WEB Services: *Apache* FTP: *Linux;/Apache* Proxy Services: *Linux/Apache* Search engine: *Swich-e*
Networking	Problem Tracking: *MultiRouter Traffic Grapher* DNS: *Linux/Bind*
Data Bases	Medium to large scale systems: Linux/Apache; MSQL; PHP/Python (LAMP Suite). Very large systems: @ TDC, using Unix, MVS. DB2, Oracle.

For Web Services….

For networking….

And for data bases……; although we still use DB2 and Oracle in

TDC's Unix and MVS environments for our larger systems.

SLIDE 15

By adopting the LAMP suite we estimate we've saved over $100,000 in one-time software costs, not to mention avoiding the ongoing costs associated with licensing and upgrades. And we'll continue to save as we develop apps in the future.

In addition to the low costs, other benefits have accrued:

- Comparable or better performance, reliability and flexibility

- Freedom from licensing hassles and inadvertent violations.

- Support from a rich variety of suppliers and user groups.

  As an aside, we had the opportunity to become a supplier after we downloaded the open source search engine Swish-e. We modified its Perl programs to meet our needs, and then posted the more powerful product on the web for others to use. With Swish-e, we can stratify our huge archive of web pages topically, enabling users to find what they're looking for quickly.

- Control over upgrades.

- Control over source code.

- User access to data not dependent on staying current with proprietary solutions. Finally, We've also achieved a better understanding and knowledge of how Internet systems work

OPEN SOURCE STRATEGIES: DESKTOP SOFTWARE

SLIDE 16

ARB's first use of open source products was to address the delivery of information. This makes sense, since the Internet was developed to support the exchange of information between disparate systems worldwide.

We then addressed the development of systems using open source products, like the LAMP suite.

The challenge now is to find suitable open source products for creating content, or information, like documents, spreadsheets, presentations, Web pages, etc. on the desktop. We need to realistically assess those features and functions our clients require, compare options, check the budget, and then determine whether there are open source solutions that will meet the need.

There are products out there: like OpenOffice, KOffice or StarOffice for the desktop. Or Sendmail or Mozilla for email. These appear to

be fully functional office applications, with a look and feel similar to Microsofts Office suite.

And they're being implemented. For example, on March 25, the Sacramento Bee ran a story on a German town that joined companies like 7-Eleven, Bears-Stearns, Pixar, Dreamworks and various governmental agencies within the US, Germany and other countries in adopting an open source office suite. According to the Bee, the small German town converted 300 PC's and 15 servers to open source products, saving $121,000 in one-time costs.

As the chart shows, the ARB could save a fair amount of money by implementing an open source office suite on the desktop.   Currently, the ARB is facing about a $850,000 bill when the time comes to upgrade its current office systems.   I'm told that our budget cannot take that kind of hit, not now, and probably not for sometime into the future.

Functions	Current Services	Possible Open Source Services	Estimated Cost Avoidance
Office Automaton Services	Email: *Netscape* Browser: *Netscape* Calendarin*g*: *Netscape*	Email: *SendMail Mozilla*	$119,000 (Cost to implement MS or Groupwise products; excludes hardware)
Desktop Apps	Office Suite: Windows 2000/Office 97	Open Source Alternatives: Linux as OS; OpenOffice; StarOffice	$734,500 (Cost to implement MS products; excludes new hardware required by apps.)

We are now testing open source options, running Linux as a desktop operating system and OpenOffice as the office suite. We'll be testing SendMail for use as an email system next fiscal year.

It seems clear to us that in the long run, if you pay nothing or very little for certain software, you almost have to save money. And often get better performance, besides. Currently, we're running Office97 on Window2000 platforms.

But who knows; if products like OpenOffice or Sendmail deliver on functionality and performance, we may adopt them. That would truly be a milestone in the use of open source products; and a serious relief from the cost and licensing issues we're all having to contend with by relying on commercial software.

FINAL THOUGHTS

Even though I probably don't need to mention this, there are a number of organizational, personnel and project management issues that will make any change successful, including rolling out open source products.

1. management support critical

2. user acceptance and involvement paramount

3. flexible IT staff

4. innovative IT managers and staff

5. internal controls over systems development

6. project planning and project management skills

But for open source solutions, you don't necessarily need a lot of money!

**SLIDE 19**

Finally, if you want more information on Open Source strategies,

products and services, check out these Web sites. They'll keep you

current on advancements and opportunities.

- Open Source Initiative: www.opensource.org

- GNU Project: www.gnu.org

- Linux: www.linuxjournal.com

- Apache Foundation: www.apache.org

- MySQL: www.mysql.com

- PHP: www.php.net

- Swish-e: http://swish-e.org/

- IBM: www.ibm.com (search on open source)

CLOSE

In closing, I hope that this overview has been useful in your

understanding of how open source options are working for at least

one state agency. They couldn't have come at a better time for us.

Now, let's take your questions. We have a panel of experts: Dr. Bill

McCarty; Harry Ng, Gary Knops, Ed P, Bill Fell, Narci Gonzales,

Susan Smalley.